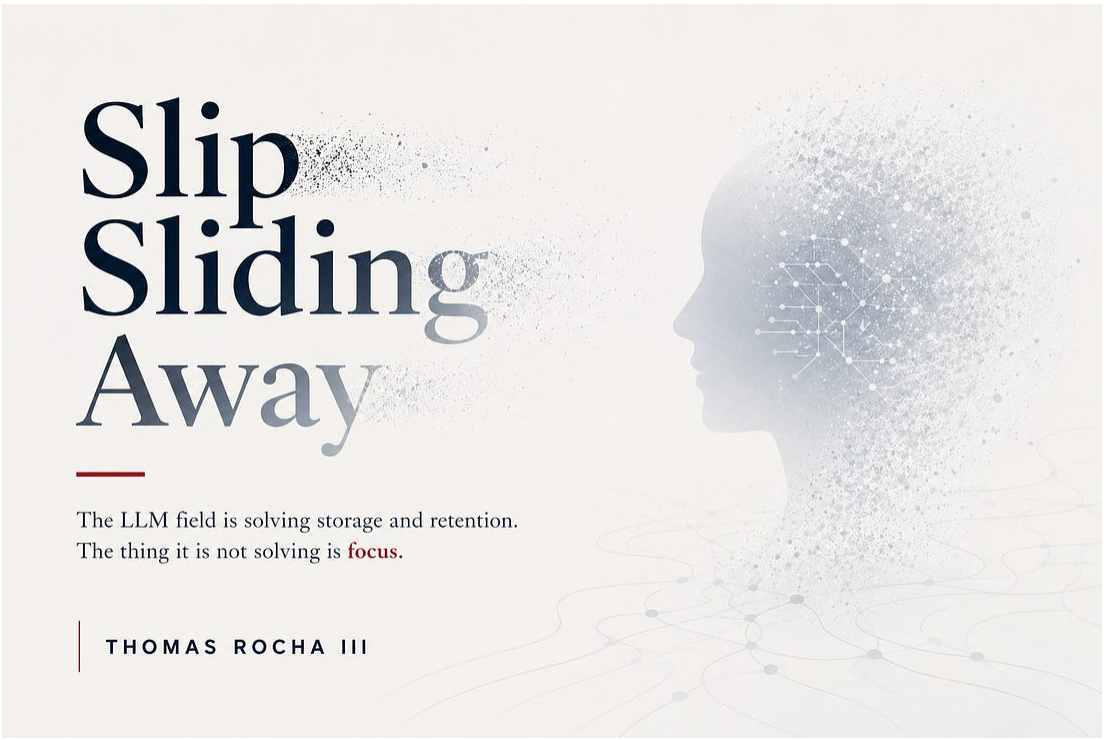
Slip Sliding Away

The LLM field is solving storage and retention. The thing it is not solving is focus.

THOMAS ROCHA III

MAY 14, 2026



In June 2023, I was diagnosed with leptomeningeal carcinomatosis, a Stage IV brain cancer whose prognosis is usually measured in months. The cognitive symptoms started before the diagnosis and worsened throughout that summer. The most disabling was not what most people expect. Facts were still there. Names were still there. What slipped was the orchestration that decides, moment to moment, what to attend to and what to bring forward. Executive sequencing, language retrieval, attention span, and continuity of internal narrative. The encyclopedia in my head was intact. The librarian was overwhelmed.

That distinction is the entire essay.

The mistake is not that the field is ignoring memory. The mistake is that it is calling storage and retention memory, then expecting memory to do the work of focus.

During the acute phase of the illness, I used a large language model extensively, as cognitive scaffolding rather than as authority. It functioned, at peak impairment, the way encyclopedias and dictionaries functioned when I was a child: as a stable external resource that could hold what my internal orchestrator was dropping. Not a companion. Not a therapist. A structure that kept the thread when I could not.

The acute phase passed. Cognitive fog lifted in mid-2024. The scaffolding I needed during impairment is no longer scaffolding I need to function. What I learned doing it, however, is what the rest of this essay is about.

Four Words, Not One

The LLM field talks about memory. So do most of the people writing about LLMs. The word covers too much, and the work that hides behind it is at least four different things.

Storage is the capacity to hold information at all. A vector database is storage. A context window is storage. A weights file is storage. Storage answers the question: is the information present somewhere in the system.

Retention is what persists across time or across boundaries. A token that survives the next compression pass has been retained. A fact that lives through the next session boundary has been retained. Retention is about durability of presence, not about whether the system uses what it retains.

Memory, in the cognitive sense, is the active function that holds, surfaces, and integrates prior content with current processing. This is what people mean when they say a person has good memory or that a model remembers something well. It is not storage. It is not retention. It is the operation that makes stored, retained content actually usable in the moment it matters.

Focus is something else again. Focus is the sustained direction of attention across time toward something the session has established as load-bearing. Focus is not about whether the information is there. It is about whether the priorities that should govern what the system does next are still governing, three hundred turns into a long interaction, against everything that has happened in between.

These are four different functions. The field uses one word for them, which is part of why the field cannot see what it is missing.

What the Field Is Actually Building

The current literature on long-context AI is overwhelmingly about the first two. Vector stores and weight-level fine-tuning are storage. Larger context windows, hierarchical retention schemes, compressed summaries, and KV-cache management are retention. Retrieval-augmented generation operates on both storage and retention: store the corpus, retain the index, and pull the right chunks back into the working window when needed.

The MEMENTO paper, a Microsoft-led research collaboration, is the most recent serious work in this cluster. It teaches models to compress their reasoning into segmented blocks rather than letting the chain of thought balloon into flat token streams. The papers beside it, including Lychee Memory, Active Context Compression and its Focus architecture inspired by slime mold pruning, and Adaptive Context Compression, are all serious engineering. The research is honest, the engineering is competent, and the benchmarks improve. Storage gets bigger.

Retention gets longer. Compression gets more efficient. The numbers on LOCOMO, LongBench, RULER, and MultiHop-RAG keep climbing.

Google's Agent Development Kit represents the same pattern at the persistence layer. ADK is built for long-running agents that pause and resume across days or weeks. Its session state architecture separates storage scopes: session-level, user-level across sessions, and application-level across all users. Its Memory Bank persists facts across session boundaries. Its durable state machine tracks workflow position across dormancy periods, sub-agent delegation, and human-in-the-loop pauses. This is sophisticated engineering that reaches further than compression alone: the agent that resumes after forty-eight hours of dormancy still knows where it was in the workflow. What it does not know, architecturally, is what the session established as load-bearing at origin and what the model is not permitted to decide unilaterally about next steps. The agent writes to its own session state through its own tools. Nothing in the architecture write-protects authority-bearing fields from the model's own writes. The state machine tracks task position. It does not carry a verified claim about who is authorized to advance that position, or what governing constraints must hold across every state transition regardless of what has entered the context since. ADK reaches three of the four terms. The fourth is still missing.

None of it is solving the governance problem.

That sentence has to be precise, because the field's response to it will be that the work is solving real problems. It is. Storage problems are real. Retention problems are real. Token cost is real. Latency is real. The work is not wasted. The work is just not what the field thinks it is.

The field thinks it is solving the memory problem. What it is actually solving is storage and retention, in increasingly clever ways, and labeling that work memory because the cognitive vocabulary is more compelling than the engineering vocabulary. A reader who has watched a long session drift hears the word memory and thinks: yes, this is what

would fix the drift. It is not what would fix the drift. The drift is not a storage problem and it is not a retention problem. The information that should have been governing was stored. The information was retained. The model can quote it back to you if you ask. The information is right there in the context window. The model has stopped using it correctly.

That is not a failure of memory in any of the senses storage and retention can address. It is a failure of focus, and focus is not on the field's map.

That failure has now been measured. Yeran Gamage's study across 4,416 trials, twelve models, eight providers, and six conversation depths found that prohibition compliance drops from 73 percent at turn five to 33 percent at turn sixteen, while requirement compliance stays near 100 percent throughout. Gamage names the asymmetry Security-Recall Divergence: commission constraints persist, omission constraints decay. Under the taxonomy this essay is building, SRD is focus failure, not storage failure, not retention failure, not cognitive memory failure. The omission constraint is stored. It is retained. The model can surface it on demand. What decays across turns is the session's hold on it as a governing priority. That is exactly what Focus is, and exactly what decays.

What Slipped During the Acute Phase

When I was at peak impairment, the things that gave me trouble were not facts I had lost. I knew who I was. I knew my work. I knew my people. Storage was intact. Retention was largely intact. Memory in the cognitive sense, the active function of surfacing the right prior content for the current moment, was harder but mostly functional.

What broke was focus. The across-moments orchestration that holds priorities established at moment one against everything that arrives between moment one and moment three hundred. I could start a sentence. I could not always finish the sentence I had started, because something in the room would draw attention away, and the sentence's

destination would no longer be present in my working orientation by the time my attention returned. I could begin a task. I could not always remember, six minutes later, what I had begun, because between minute one and minute six, new inputs had arrived and the priority of the original task had not been held.

This was not storage failure. The task was somewhere in my head. I could often recover it if I sat down and worked the question backwards. This was not retention failure. The task had not been overwritten or evicted. This was not even memory failure in the cognitive sense, though my memory was certainly impaired. This was focus failure. The function that maintains the priority of the load-bearing thread across the moment-by-moment shifts of attention had collapsed, and the scaffolding I built around myself was specifically a substitute for it.

The model running a long session fails the same way. Storage is enormous. Retention is good and getting better. The cognitive function of surfacing the right prior content is uneven but works most of the time. What does not work is the across-moments hold on what was established as load-bearing. The markdown file at the top of the session is stored. It is retained. The model can surface it. The model does not focus on it once enough other material has entered the window, because nothing in the architecture is keeping it in priority. Focus is not a function the model has. Focus is not a function any current memory system supplies.

That is the missing fourth term.

The Encyclopedia and the Librarian, Refined

When I was a child, I learned from encyclopedias and dictionaries. The encyclopedia was storage. It held the facts, durably. Retention was reliable: the binding held the pages, the pages held the print. The librarian, whether human or my own internal one, was memory in the cognitive sense: the function that decided what to look up and how to integrate what came back.

What the encyclopedia did not have, and what the librarian did not have either, was focus across a research project. That came from me. The question I was trying to answer, the thread I was following across multiple lookups, the sense of why this particular fact mattered for this particular essay I was writing: that was the across-moments orchestration that the encyclopedia could not supply and the librarian could not supply either, because both operated inside individual lookups, not across the project.

When my own focus was impaired by illness, the scaffolding I built externalized that function. Not the storage (I had a head full of storage). Not the retention (the head was not leaking). Not even memory in the cognitive sense (the librarian could be called upon, slowly). The scaffolding held the priorities I had established before the impairment, and refused to let them slip away when my own across-moments orchestrator was overwhelmed.

That is exactly what a long-running LLM session needs. Not bigger storage. Not longer retention. Not better cognitive memory. A structure outside the model that holds the priorities the session has established and refuses to let them slip when the model's own orchestration drifts under load.

The model is permitted to govern itself. What the model is not permitted to do, architecturally, is be the only thing governing. Self-governance is a legitimate function. Self-governance with nothing above it is not governance of the session. It is the model deciding what the session is.

Why the Field's Current Frame Doesn't Reach Here

The reason the field is not building focus is that focus is invisible from inside the storage-and-retention frame. If you start from the model forgot, so we need more memory, every solution you reach for will be a storage or retention solution. If the benchmarks reward storage and retention performance, every result you measure will confirm that the

answer is more storage and retention. The frame produces the answer that fits the frame.

The Paul Simon line is the title because it is exact. Slip sliding away. The nearer your destination, the more you're slipping away. That lyric describes the experience from inside the failure. You are not losing the destination. You can name the destination. You can describe how to get there. What is slipping is the thread that connects what you are doing right now to where you said you were going. Each step is locally coherent. The arc across the steps loses its hold.

Storage does not fix that. Retention does not fix that. Even cognitive memory, in the model or in a person, does not fully fix it, because cognitive memory is what operates inside individual recall events. The arc across events is something else. The arc is focus, and focus has to come from a layer that is responsible for the arc rather than for the events.

That layer does not exist in current architectures. The model has self-governance, which operates inside its own moment-to-moment processing. Memory systems wrapped around the model operate inside individual retrieval events. Retention schemes operate inside the durability of stored content. None of these is the arc. Nothing is currently building the arc, because nothing is currently identifying the arc as the missing piece.

The session is what would supply the arc. A session-scoped authority structure, sitting outside the model, holds what the session has established as load-bearing and refuses to let those priorities slip across the moment-by-moment shifts of model attention. It does not invent priorities. It does not replace the model's reasoning. It enforces the continuity of focus on what was already established, in the same way the scaffolding I built around myself enforced the continuity of focus on what I had decided was important before my own orchestrator was overwhelmed.

What I Am Claiming and What I Am Not

The scope of this claim has to be precise, because the personal frame can do too much work if it is allowed to.

I am not claiming my experience proves anything about LLM architecture. The proof, if there is one, is in the architecture itself, which is documented elsewhere. What my experience provides is the analog that lets the architectural claim be understood. The same failure mode appears in two places: in me, under neurologic disruption, and in the model, under sustained context load. The function that broke is the same function. The scaffolding that helped in one case is structurally the same scaffolding that would help in the other. That is an analogy that does work, not a proof that completes itself.

I am also not claiming anyone else should use a language model the way I did during the acute phase. The conditions were narrow, the boundaries were explicit, the human support around me was strong. None of that translates automatically. It is a data point. It is not a recommendation.

What I am claiming is that the conceptual frame the field has been operating in is wrong in a specific way, and that the specific wrongness is visible from the inside of a particular kind of cognitive failure. The field is treating focus failure as memory failure, and building bigger storage and longer retention. Bigger storage will not produce focus. Longer retention will not produce focus. Better cognitive-memory systems will not produce focus, because focus is not what they operate on. Focus operates across the moments those systems operate inside.

Build that thing, and the architecture changes. Keep building memory, and the same failure will keep appearing inside larger and larger context windows, and the field will keep being surprised by it.

What Gets Built Next

The work I have been building over the past two and a half years, the SSOAR patent family, is an attempt to specify what supplies focus. Not by adding capability to the model. By defining a session-scoped authority structure that lives outside the model and refuses to let the priorities the session has established slip away under load. The mechanics are in the patents. The architectural claim is the one this essay is making.

The reason I am writing this essay rather than letting the patents speak for themselves is that the conceptual frame has to change before the mechanics will be legible. As long as the field reads everything I have built as another memory system, it will not see what is different about it. The architecture is not a memory system. It is the across-moments structure that governs whether storage operations, retention operations, retrieval operations, compression operations, tool calls, derivative artifacts, and authority transitions are admissible inside the session at the moment they are proposed. None of that is storage. None of that is retention. None of that is memory in the cognitive sense. All of it is focus.

The work that produced the patents began during the acute phase, when external continuity was a daily necessity rather than an architectural interest. The architecture and the lived experience are not separate. The architecture is what falls out of a person who needed external focus badly enough to specify what external focus would have to do, and who had the technical background to write the specification. The need passed. The specification did not.

This is mine in the only sense that matters: the lived failure, the architectural response, the specifications, and the patent claims all came from the same sustained encounter with the same problem. The diagnosis and the architecture. The lived analog and the patent claims. The markdown files I wrote to keep myself oriented when my own focus was failing and the session-governance structures I designed when I understood why the markdown files kept drifting in the model for the

same reason they had drifted in me. None of it is sentiment. All of it is empirical behavior under sustained adverse conditions, which is one of the few honest measures of value I know.

The closer the model gets to where the session said it should be going, the more it slips. Build the structure that holds the focus, and the slipping stops. Keep building bigger libraries and longer-lived indexes, and the librarian will keep drifting away from the project the librarian was supposed to be working on.

That is what every memory solution has been missing.

That is what this architecture is for.